Boundary detection using quadratic filters: performance criteria and experimental assessment

Pietro Perona, Jitendra Malik

Pietro Perona, Jitendra Malik, "Boundary detection using quadratic filters: performance criteria and experimental assessment," Proc. SPIE 1708, Applications of Artificial Intelligence X: Machine Vision and Robotics, (1 March 1992); doi: 10.1117/12.58583

SPIE.

Event: Aerospace Sensing, 1992, Orlando, FL, United States

Boundary detection using quadratic filters: performance criteria and experimental assessment*

Pietro Perona[1,2] *and Jitendra Malik*[3]

[1] California Institute of Technology 116-81, Pasadena CA 91125, USA
[2] Università di Padova, Dipartimento di Elettronica ed Informatica
[3] University of California at Berkeley, Evans Hall, Berkeley CA 94720

Abstract. It is well known that the projection of depth or orientation discontinuities in a physical scene results in image intensity edges which are not ideal step edges but are more typically a combination of steps, peak and roof profiles. However most edge detection schemes ignore the composite nature of intensity edges, resulting in systematic errors in detection and localization. We have addressed [14] the problem of detecting and localizing these edges, while at the same time also solving the problem of false responses in smoothly shaded regions with constant gradient of the image brightness. We have shown that a class of nonlinear filters, known as quadratic filters are appropriate for this task, while linear filters are not. In this paper a series of performance criteria are derived for characterizing the SNR, localization and multiple responses of these quadratic filters in a manner analogous to Canny's criteria for linear filters. Additionally we show experiments on a series of images varying systematically the parameters of the edge detector.

1 Introduction

The problem of detecting and localizing discontinuities in greyscale intensity images has traditionally been approached as one of finding step edges. This is true both for the classical linear filtering approaches as well as the more recent approaches based on surface reconstruction.

Unfortunately, step edges are an inadequate model for the discontinuities in the image that result from the projection of depth or orientation discontinuities in physical scene. Mutual illumination and specularities are quite common and their effects are particularly significant in the neighborhood of convex or concave object edges. In addition, there will typically be a shading gradient on the image regions bordering the edge. As a consequence of these effects, real image edges are not step functions but more typically a combination of steps, peak and roof profiles (see [14] Figure 1). This had been noted experimentally by Herskovits and Binford back in 1970. Quantitative analyses of the associated physical phenomena have also been provided- Horn[7] and more recently Forsyth and Zisserman [4].

Most local edge detection methods are based on some decision making stage following a linear filtering stage. Typically one looks for maxima in the filtered image perpendicular to the orientation of the edge. Such an approach (e.g. Canny [3])results in a systematic error in localization whenever there is a composite edge([15](page 9), or [2](Fig. 2. 1)). In [14] we show that this problem is not specific to the Gaussian derivative filters used by Canny, but is present whatever the linear filter used. For any such filter there is a systematic localization error for composite edges. Using any (finite) number of linear filters does not help. However, a *quadratic* filtering approach as proposed by Burr, Morrone, Owens and their colleagues [9, 8] is adequate. Instead of looking for maxima in $(I * f)$ one looks for maxima in $W = (I * f_1)^2 + (I * f_2)^2$, or more generally $\sum (I * f_i)^2$. One can prove [14] that if f_1 and f_2 are properly scaled one can have zero systematic error (see also section 2.6).

If one is to design an 'optimal' quadratic filtering approach, one needs to formulate computable forms of design criteria, analogous to the ones used by Canny [3] for linear filtering. We report here such calculations (sec. 2). We also describe the 2D implementation of the edge-detector and a number of experiments on real images.

2 Computation of the performance criteria

In the choice of a filter one would like to minimize different types of edge-detection errors. What follows is a list of criteria for evaluating quadratic filtering-based edge-detectors.

* This research was partially supported by NSF Presidential Young Investigator Award IRI-8957274 to J.M. and by U.S. Army Research Office grant number DAAL 03-86-K-0171 (P.P.).

Signal to noise ratio (Sec. 2.2) – Ratio of signal response to the variance of the response due to noise.
Multiple edges (Sec. 2.3) Edges not present in the signal found in the neighbourhood of real edges, due to multiple ripples in the convolution kernels.
Spurious edges (Sec. 2.4) Edges detected far from 'true' edges. Due to response to high brightness gradient regions.
Deterministic localization error (Sec. 2.5) – Error committed in locating the edge in the no-noise situation.
Stochastic localization error (Sec. 2.6) – Localization error due to noise.
Multiple responses (Sec. 2.7) – Edges detected in the neighbourhood of a true one due to noise in the data.
Multiple responses (Sec. 2.8) – Edges detected far from the 'true' edges. Due to response to noise.

After establishing some notation this section is devoted to making a quantitative assessment of these criteria.

2.1 Notation

Edge – $G_c(x) = c_1\delta^{(-2)}(x) + c_2\delta^{(-1)}(x) + c_3\delta(x)$, $\mathbf{c}^T = [c_1, c_2, c_3]$
Noise – $N(x) = n_0\eta(x)$, $\eta(x)$ being white zero-mean unit-variance Gaussian noise.
Image – $I(x) = G_c(x) + N(x)$ - Signal + noise.
Kernels – $\mathbf{f}(x)^T = [f_1(x), \ldots, f_n(x)]$, and, for convenience, $\mathbf{F}(x)^T = [F(x)_1, \ldots, F(x)_n]$, with $\mathbf{F}''(x) = \mathbf{f}(x)$
Responses – $\mathbf{r}_{G_c}(x) = (\mathbf{f} * G)(x)$; $\mathbf{r}_N(x) = (\mathbf{f} * N)(x) = n_0\mathbf{r}_\eta(x)$; $\mathbf{r}(x) = \mathbf{r}_{G_c}(x) + \mathbf{r}_N(x)$
Power – $W_{G_c}(x) = \|\mathbf{r}_{G_c}(x)\|^2$; $W(x) = \|\mathbf{r}(x)\|^2$
Correlations – The nxn correlation matrix is defined componentwise by: $\mathbf{R}_{fg(ij)}(t) \doteq \langle f_i(\cdot + t), g_j(\cdot)\rangle_{L_2}$ For simplicity of notation whenever $f = g$ we will write $\mathbf{R}_f$ instead of $\mathbf{R}_{ff}$.
Kernel derivatives – $\mathbf{H}$ defined componentwise by $\mathbf{H}_{ij} = F_i^{(j-1)}$, $j = 1, \ldots, 3$ and $i = 1, \ldots, n$.

2.2 Signal to noise ratio

Define signal to noise ratio as the ratio of the response to pure signal at the edge and the standard deviation of the response to pure noise. Using our notation:

$$SNR \doteq \frac{\|\mathbf{r}_{G_c}(0)\|}{\sqrt{E\|\mathbf{r}_N\|^2}} \tag{1}$$

The variance of the response to pure noise depends on the correlation matrix of the functions f(x):

$$E\|\mathbf{r}_N\|^2 = E\sum_i (f_i * N)^2 = n_0^2 \sum_i \langle f_i, f_i\rangle = n_0^2 tr(\mathbf{R}(0)) \tag{2}$$

For a generic edge image $G_c(x)$ the signal to noise ratio is then:

$$SNR = \frac{\|\mathbf{r}_{G_c}(0)\|}{n_0\sqrt{tr(\mathbf{R}(0))}} \tag{3}$$

In the special case that the edge is a combination of roof, step and line: $G = c_1\delta^{(-2)} + c_2\delta^{(-1)} + c_3\delta$ the signal to noise ratio becomes:

$$SNR = \frac{\|\mathbf{Hc}\|}{n_0\sqrt{tr(\mathbf{R}(0))}} \tag{4}$$

2.3 Ripples in the filters

Our strategy for detecting edges is to look for maxima in the filter response. It is important therefore that the resposnse of the filters in the no-noise situation, $W(x) = W_{G_c}(x)$, has only one peak per edge. Therefore a requirement that we make is that $W_{G_c}(x)$ is unimodal, independently of the composition of the edge, i.e. that (call x_e the coordinate of the maximum)

$$\frac{\partial}{\partial x} W_{G_c}(x) > 0 \qquad x < x_e \qquad \forall c \tag{5}$$

$$\frac{\partial}{\partial x} W_{G_c}(x) < 0 \qquad x > x_e \qquad \forall c \tag{6}$$

Since $W_{G_c} = \|\mathbf{Hc}\|^2$ the condition is equivalent to the matrix $\mathbf{H}'^T\mathbf{H}$ being positive definite for $x < x_e$ and negative definite for $x > x_e$ (cfr the condition that the matrix is antisymmetric in zero, sec. 2.5).

2.4 Spurious edges due to shading gradients

A well known problem of first derivative edge detectors is that they respond with false edges in areas with smooth shading even when the gradient of brightness is constant. To avoid these false positives, one may have to set a threshold which leads to the rejection of genuine low–contrast edges. This problem has persisted in the 'modern' approaches based on surface reconstruction. Whether the formulation is a probabilistic one using MRFs (e.g. Geman and Geman) or a variational one, if the cost function includes terms like the squared gradient there will be a tendency towards piecewise constant reconstructions. Blake and Zisserman [2] even provide a computation of the gradient threshold above which false edges get created.

In the linear filtering framework, Binford [1] describing the Binford–Horn line finder discusses one solution to this problem– a lateral inhibition stage preceding the stage of finding directional derivatives. Essentially this amounts to using third derivatives, and suffers from the expected weakness–low signal to noise ratio compared to first derivative operators. A simple calculation using the SNR criterion defined by Canny [3] confirms this.

A compact characterization of filters which do not suffer from the linear gradient problem can be obtained as follows: suppose that the image just consists of a ramp function $I(x) = \delta^{(-2)}(x)$. The response of a linear filter f to such a ramp is $I * f = f^{(-2)}(x)$. It can be seen that $f^{(-2)}(x)$ should satisfy the following two conditions:

1. $\|f^{(-2)}(x)\| \rightarrow 0$ for $\|x\| \rightarrow \infty$. This ensures that far enough from the roof junction, the response to a ramp is negligible.
2. $f^{(-2)}(x)$ either has a zero crossing or a maximum or a minimum at the origin. This is to enable the localization of onset of the ramp without any bias.

While the third derivative of a Gaussian $G'''_\sigma(x)$ is one filter which would satisfy these criteria, there are others which do so without that significant a drop in SNR. One such choice is the Hilbert Transform of $G''_\sigma(x)$ which is an odd-symmetric filter. We computed Canny's SNR and localization criteria for this filter and compared it with $G'_\sigma(x)$. It turns out that for $G'_\sigma(x)$, the SNR is $1.062\sigma^{0.5}$ and localization is $0.8673\sigma^{-0.5}$. For $(G''_\sigma)_H(x)$, the SNR is $0.6920\sigma^{0.5}$ and localization is $0.87535\sigma^{-0.5}$. Considering the product of the SNR and the localization, the numbers are 0.92 and 0.606 respectively implying that $(G''_\sigma)_H$ is worse by about 34%. However, its r value is 0.676 which is 32 % better than $r = 0.51$ for the G'_σ. In other words, while the $(G''_\sigma)_H$ is roughly comparable to the G'_σ filter used by Canny, its immunity to smooth shading makes it preferable.

This observation is independent of our other main concern in this paper–that of correctly dealing with composite edges–and could be directly exploited. One caveat: Since this filter has additional side lobes, the reponse to an ideal step edge shows a ringing like phenomenon. In this paper we use such a filter as part of a quadratic filter, this ringing does not appear (See [14] Fig. 2. Note how in each case, the composite edge is correctly localized and that the filter is insensitive to linear shading.)

2.5 Systematic localization error – Normalization conditions

In the scheme we propose edges are detected whenever $W(x)$ reaches a maximum. A natural and simple criterion for localizing the position of each detected edge is to define it to be the coordinate $x = x_e$ where W reaches the maximum. This localization criterion is quite arbitrary: given a generically shaped edge signal it is not clear where the edge should be located. In some special cases, however, the position of the boundaries is naturally defined. Consider a signal G defined as in section 2.1; whatefer the choice of the coefficients $\mathbf{c}$, the edge is located at $x = 0$, so $x_e - 0 = x_e$ is a localization error. In this section x_e is computed for a given choice of twice continuously differentiable filters $\mathbf{f}$ when the noise is zero, i.e. $W = W_{G_c}$ (in the rest of this section we will write W instead of W_{G_c}).

A necessary and sufficient condition for x_e to be a maximum point is that $W'(x_e) = 0$ and $W''(x_e) < 0$. Expanding W' in Taylor sum around $x = 0$ and computing it in $x = x_e$ we obtain:

$$0 = W'(x_e) = W'(0) + W''(0)x_e + \mathcal{O}(x_e^2) \tag{7}$$

which gives us an estimate of x_e in terms of the derivatives of W at the origin:

$$x_e \approx \frac{W'(0)}{-W''(0)} \tag{8}$$

The derivatives of W may computed yelding:

$$x_e \approx -\frac{\mathbf{c}^T \mathbf{H}'^T(0)\mathbf{H}(0)\mathbf{c}}{\mathbf{c}^T \mathbf{H}''^T(0)\mathbf{H}(0)\mathbf{c} + \|\mathbf{H}'(0)\mathbf{c}\|^2} \tag{9}$$

A necessary and sufficient condition for the systematic localization error to be zero is therefore that the filter collection $\mathbf{f}$ satisfies the conditions:

$$\mathbf{c}^T \mathbf{H}'^T(0)\mathbf{H}(0)\mathbf{c} = 0 \qquad \forall \mathbf{c} \in \mathbb{R}^3 \tag{10}$$

$$\mathbf{c}^T \mathbf{H}''^T(0)\mathbf{H}(0)\mathbf{c} < -\|\mathbf{H}'(0)\mathbf{c}\|^2 \qquad \forall \mathbf{c} \in \mathbb{R}^3 \tag{11}$$

Notice that condition (10) is equivalent to $\mathbf{H}'^T(0)\mathbf{H}(0)$ being an antisymmetric matrix. In fact any matrix $\mathbf{A}$ may be written as the sum of its symmetric and antisymmetric components $2\mathbf{A} = 2\mathbf{A}_s + 2\mathbf{A}_a \doteq (\mathbf{A} + \mathbf{A}^T) + (\mathbf{A} - \mathbf{A}^T)$. The product $\mathbf{c}^T\mathbf{A}\mathbf{c}$ is equal to the product involving the symmetric part only: $\mathbf{c}^T\mathbf{A}\mathbf{c} = \mathbf{c}^T\mathbf{A}_s\mathbf{c}$. The symmetric part may be diagonalized ($\mathbf{A}_s = \mathbf{T}^T\mathbf{D}\mathbf{T}$) giving $0 = \mathbf{c}^T\mathbf{A}\mathbf{c} = (\mathbf{T}\mathbf{c})^T\mathbf{L}(\mathbf{T}\mathbf{c}) \Leftrightarrow \mathbf{L} = 0 \Leftrightarrow \mathbf{A}_s = 0 \Leftrightarrow \mathbf{A} = -\mathbf{A}^T$.

Equation (10) is then equivalent to:

$$\mathbf{H}'^T(0)\mathbf{H}(0) = -\mathbf{H}^T(0)\mathbf{H}'(0) \tag{12}$$

which may be written componentwise as:

$$\sum_k \mathbf{H}'_{k,i}\mathbf{H}_{k,j} = -\sum_k \mathbf{H}_{k,i}\mathbf{H}'_{k,j} \tag{13}$$

From the definition of $\mathbf{H}$:

$$\sum_{k=1}^{n} F_k^i(0)F_k^{(j-1)}(0) = -\sum_{k=1}^{n} F_k^j(0)F_k^{(i-1)}(0) \qquad i,j = 1\ldots3 \tag{14}$$

Observe that when $\mathbf{f}^T = [f_1, f_2] = [f_e, f_o]$ with f_e and even kernels and f_o an odd kernel, the equations 14 specialize as follows: (a) when $i+j$ is even the equations are trivially satisfied since they are formed of products $F_i^r F_i^t$ with $s+t$ odd, and since F_e has odd derivatives equal to zero in zero and F_o has even derivatives equal to zero in zero all such products are zero. (b) when $i+j$ is odd then the equations are:

$$(i=1, j=2), (j=1, i=2) \qquad F'^2_2(0) = -F''_1(0)F_1(0) \tag{15}$$

$$(i=2, j=3), (j=2, i=3) \qquad F''^2_1(0) = F'''_2(0)F'_2(0) \tag{16}$$

Equation (16) is obtained in the step + delta edge model case, while Equation (15) is added by the introduction of the ramp. In conclusion: when f_e and f_o are such that Eq. (16) and Eq. (15) are satisfied there will be no systematic localization error for any edge which is a linear combination of a step, delta and ramp, when f_e and f_o are such that Eq. (15) is satisfied there will be no systematic localization error for any edge which is a linear combination of a step and a delta. Two observations are of importance: (a) While Eq. (15) may be easily satisfied by renormalizing appropriately a previously chosen pair of kernels, the pair of equations (15) and (16) is in general more stringent. (b) The fact that we have used a ramp, step and delta to build our model of an edge somehwat simplifies the calculations, but it is not of great importance. For example an exponential $exp(-\tau\|x\|)$ could be used in place of the delta function, and the calculations would carry through in the same way.

2.6 Stochastic localization error

In this section we study the localization error due to noise added to the image. We label $x = 0$ the position where the response $W_{G_c}(x)$ to the noiseless signal has a peak (i.e. $W'_{G_c}(0) = 0$), and $x = x_0$ the coordinate where the response to noisy signal $W(x)$ does (i.e. $W'(x_0) = 0$). We expand the derivative $W'(x)$ of the response in power series around the $(x = 0, n_0 = 0)$ point, and compute it at the (x_0, n_0) point.

The derivative of W in x_0 is:

$$W'(x_0) = W'_{G_c}(x_0) + 2n_0\left(\mathbf{r}^T_{G_c}\mathbf{r}_\eta\right)'(x_0) + \mathcal{O}(n_0^2) = 0 \tag{17}$$

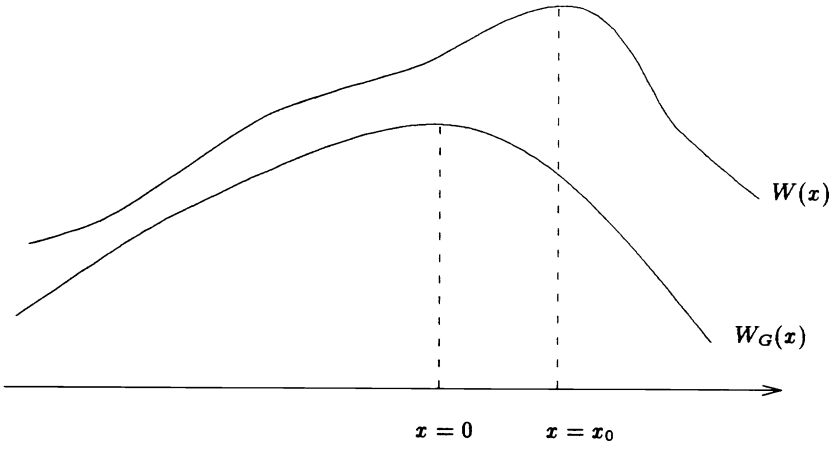


Fig. 1. Localization error due to noise.

The first two terms of the second member of equation (17) can be expanded in Taylor sums around the origin:

$$W'_{G_c}(x_0) = W'_{G_c}(0) + W''_{G_c}(0)x_0 + \mathcal{O}(x_0^2) = W''_{G_c}(0)x_0 + \mathcal{O}(x_0^2) \tag{18}$$

$$\left(\mathbf{r}_{G_c}^T \mathbf{r}_\eta(x_0)\right)' = \left(\mathbf{r}_{G_c}^T \mathbf{r}_\eta\right)'(0) + \left(\mathbf{r}_{G_c}^T \mathbf{r}_\eta\right)''(0)x_0 + \mathcal{O}(x_0^2) \tag{19}$$

Substituting equations (18), (19) into equation (17) and solving for x_0 we obtain:

$$x_0 \approx n_0 \frac{2\left(\mathbf{r}_{G_c}^T \mathbf{r}_\eta\right)'(0)}{-W''_{G_c}(0) + \mathcal{O}(n_0)} \approx n_0 \frac{2\mathbf{r}'^T_{G_c}\mathbf{r}_\eta(0) + 2\mathbf{r}_{G_c}^T \mathbf{r}'_\eta(0)}{-W''_{G_c}(0)} \tag{20}$$

The expectation of the stochastic localization error is clearly zero, and the variance may be approximated using (20):

$$Ex_0^2 \approx n_0^2 \frac{E(\mathbf{r}_{G_c}^T \mathbf{r}_\eta(0))^2}{W''_{G_c}(0)^2} = n_0^2 \frac{\mathbf{r}'^T_{G_c}\mathbf{R}_f \mathbf{r}'_{G_c}(0) + 4\mathbf{r}'^T_{G_c}\mathbf{R}_{ff'}\mathbf{r}_{G_c}(0) + \mathbf{r}_{G_c}^T \mathbf{R}_{f'f'}\mathbf{r}_{G_c}(0)}{W''_{G_c}(0)^2} \tag{21}$$

2.7 Spacing of the maxima in the neighbourhood of an edge

We will suppose that the noise variance, n_0, is small with respect to the magnitude of the signal. We will therefore approximate the value of $W(x)$ disregarding term quadratic in n_0:

$$W = W_{G_c} + 2n_0 \mathbf{r}_{G_c}^T \mathbf{r}_\eta + n_0^2 W_\eta \approx W_{G_c} + 2n_0 \mathbf{r}_{G_c}^T \mathbf{r}_\eta \tag{22}$$

We may apply a generalization (see [13]) of Rice's formula [17] to compute the expected value of the distance between maxima of the random process $W_\Delta(x) = W(x) - W_{G_c}(x) = 2n_0 \mathbf{r}_{G_c}^T \mathbf{r}_\eta$. Notice that the expectation has an argument x since it depends on the distance from the location of the edge $G_c(x)$. In a neighbourhood of the edge we expect the derivative of W_G to be close to zero and thus the estimate of the spacing of the maxima of $W_\Delta(x)$ to be a good estimate of the spacing of the maxima of $W(x)$. For the sake of simplicity we evaluate the function in $x = 0$ (in principle one should verify that $x = 0$ is the worst case):

$$d_{W_\Delta}(0) = 2\pi \frac{\sqrt{a(0)c(0) - b^2(0)}}{c(0) - b(0)} \tag{23}$$

Where $a(x)$, $b(x)$ $c(x)$ in $x = 0$ are (see [13]):

$$\begin{aligned}
a(0) &= \mathbf{r}'_{G_c}(0)^T \mathbf{R}_f \mathbf{r}'_{G_c}(0) + \mathbf{r}_{G_c}(0)^T \mathbf{R}_{f'} \mathbf{r}_{G_c}(0) \\
b(0) &= \mathbf{r}'_{G_c}(0)^T \mathbf{R}_f \mathbf{r}''_{G_c}(0) + \mathbf{r}'_{G_c}(0)^T \mathbf{R}_{ff''} \mathbf{r}_{G_c}(0) + 2\mathbf{r}_{G_c}(0)^T \mathbf{R}_{f'} \mathbf{r}'_{G_c}(0) \\
c(0) &= \mathbf{r}''_{G_c}(0)^T \mathbf{R}_f \mathbf{r}''_{G_c}(0) + 4\mathbf{r}'_{G_c}(0)^T \mathbf{R}_{f'} \mathbf{r}'_{G_c}(0) + \mathbf{r}_{G_c}(0)^T \mathbf{R}_{f''} \mathbf{r}_{G_c} + 2\mathbf{r}''_{G_c}(0)^T \mathbf{R}_{ff''} \mathbf{r}_{G_c}(0)
\end{aligned} \tag{24}$$

2.8 Spacing of the maxima far from an edge

We have not been able to compute this in closed form a function of the kernels' shape. One may take the sum of the estimates of the number of maxima of $f_k * \eta$ as an upper bound to this number, therefore, using Rice's formula [17, 3]:

$$T_{[1..n]} = 2\pi \sum_{k=1}^{n} \sqrt{\frac{R_{f'_k}(0)}{R''_{f'_k}(0)}} \tag{25}$$

3 Detecting edges in two dimensions

To detect edges in 2D, we use a Gaussian window to compute the 2D extension of the 1D kernels discussed so far. Rotated copies of the kernels are used to (conceptually) compute a response $R(x, y, \theta)$. At each point, the locally dominant orientations θ_i which correspond to the local maxima (over θ) are determined. Allowing for multiple orientations enables junctions to be detected consistently. Edge points are defined as the points where the directional derivative in the direction perpendicular to a locally dominant orientation is 0.

In practice one cannot afford to compute convolutions of the image with kernels at an infinity of orientations. It turns out that it is possible to approximate $R(x, y, \theta)$ with arbitraty precision using linear combinations of a finite number of functions as described by Freeman and Adelson [5, 6] and Perona [10, 11, 12]. It is therefore possible to compute $R(x, y, \theta)$ on a continuum of orientations (see Fig.2-b).

3.1 Edge detection

At edge points the filter output 'energy' R (see Fig.2-b) will have a maximum at the orientation θ_e parallel to the edge (see Fig.2-c). Fix θ_e and consider $R(x, y, \theta_e)$. Along a line orthogonal to the edge the problem reduces to the 1D case: there will be an energy maximum at the edge. Points through which edges pass can be found by marking as 'edge points' all the points $p = (x, y, \theta)$ that satisfy:

$$\frac{\partial}{\partial \theta} R(p) = 0 \tag{26}$$

$$\frac{\partial}{\partial \mathbf{v}_\theta} R(p) = 0 \tag{27}$$

where $\mathbf{v}_\theta$ is the unit vector orthogonal to the orientation associated to θ.

The search for the edge points has been implemented as follows:

1. For each image pixel (x, y) the angles $\theta_i(x, y)$ at which the response is maximized are found. For this operation we use Brent's maximization algorithm (See [16]). The upper bound on the orientation error was set at 1 degree (see Fig.2-c). The angle space is coarsely sampled (approx. a sample every 5 degrees) to provide initial conditions for the bracketing algorithm. The energies $R_i(x, y)$ corresponding to $\theta_i(x, y)$ are also stored. The lower 70% of the sampled energies at each point are averaged to give a global noise estimate.
2. The maxima in (x, y) of $R(x, y, \theta_i(x, y))$ are computed with sub-pixel accuracy by fitting a cylindrical paraboloid to $R(x, y, \theta_i)$ in the 3x3 neighbourhood around the pixel (See Fig.2-d). Points where the fit error (see Eq. (38) is greater than a preset threshold value are discarded. A typical threshod we used was 15%. See section 3.2 for the details.
3. The edge pixels are thresholded. The thresholding is based on a threshold based on an estimate of the average noise.

The results of the search are shown in figure 2-d and in the experimental section are compared to the output of a Canny detector using filters of the same scale. The typical number of steps of the Brent algorithm was 2-3. The time spent in the search was always less than the filtering time.

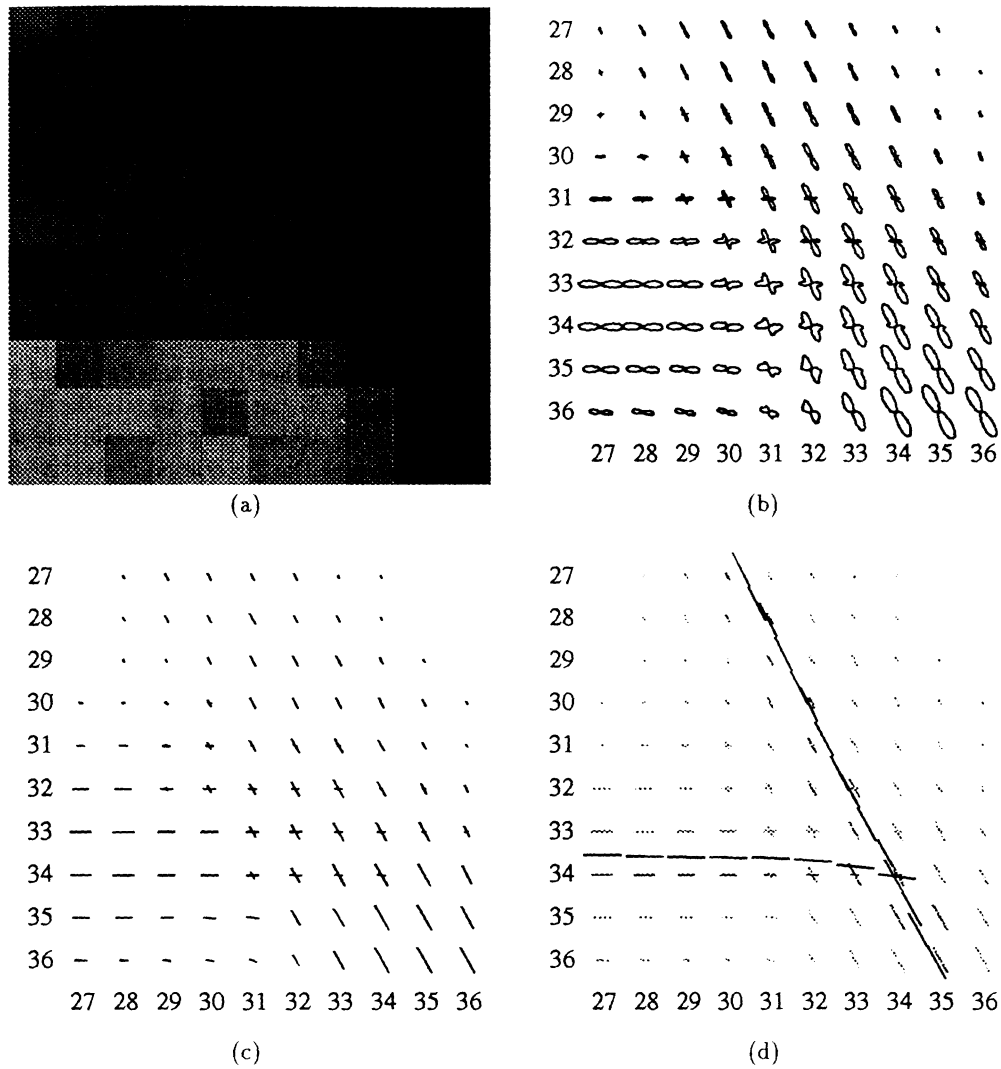


Fig. 2. The process of brightness edge detection shown for a detail (a) of the image of a T junction. (b) Energies are computed at every pixel as a function $R(x, y, \theta)$. At each pixel the energy is represented by a polar plot; the plots are π-periodic since $R(x, y, \theta)$ does not encode the direction, only the orientation of the edges. Notice that the energies have (c) Maxima in θ are calculated at each pixel. The length of the needles indicates the associated energy. (d) 'Oriented' maxima in (x, y) are computed at subpixel resolution.

3.2 Sub-pixel localization of the edges

As we have discussed above, we associate the edges in the image with the 'ridges' of a function $W(x, y, \theta)$ (see Eq.(26). The W function is computed on the nodes of a lattice corresponding to the data lattice. This need not limit the accuracy with wich the position and the shape of the ridges can be determined. The function W has a very regular behaviour and may be interpolated to any point of a continuum in x, y, θ.

In this section we describe an interpolation method based on fitting a second order model to the 'dorsals' of the ridges of W at a certain angle θ. The model is a paraboloid whose axis of symmetry has an angle θ with the X axis, and is described by:

$$z = ad^2 + bd + c \qquad d(x, y) = -xsin\theta + ycos\theta \tag{28}$$

The model has three parameters a, b, c. The vector $[a, b, c]^T$ will be referred to as $\mathbf{a}$. Notice that the model is linear in $\mathbf{a}$.

Consider a 3x3 neighbourhood. The nine data points may be expressed in terms of the model in matrix form:

$$\mathbf{z} = A\mathbf{a} \qquad \mathbf{z}^T = [y_1, \ldots, y_9] \qquad A_{i,j} = d_i^j \qquad i = 1, \ldots, 9; j = 0, \ldots, 2 \tag{29}$$

with d defined as above.

The linear least squares estimator for $\mathbf{a}$ is then (see e.g. [16]):

$$\hat{\mathbf{a}} = (A^T A)^{-1} A^T \mathbf{z} \tag{30}$$

The matrix $(A^T A)^{-1}$ may be computed explicitly. Define the quantity $\beta = 2sin^2\theta cos^2\theta + 1$. Then:

$$A^T A = \begin{bmatrix} 6\beta & 0 & 6 \\ 0 & 6 & 0 \\ 6 & 0 & 9 \end{bmatrix} \tag{31}$$

and defining $\Delta = 6(3\beta - 2)$:

$$(A^T A)^{-1} = \Delta^{-1} \begin{bmatrix} 3 & 0 & -2 \\ 0 & 3\beta - 2 & 0 \\ -2 & 0 & 2\beta \end{bmatrix} \tag{32}$$

Therefore the operator associated to the estimator is:

$$(A^T A)^{-1} A^T = \Delta^{-1} \begin{bmatrix} 3d_1^2 - 2 & 3d_2^2 - 2 & \dots & 3d_9^2 - 2 \\ (3\beta - 2)d_1 & (3\beta - 2)d_2 & \dots & (3\beta - 2)d_9 \\ -2d_1^2 + 2\beta & -2d_2^2 + 2\beta & \dots & -2d_9^2 + 2\beta \end{bmatrix} \tag{33}$$

So the parameters have the following closed-form estimators:

$$\hat{a} = \Delta^{-1} \sum_{i=1}^{9} (3d_i^2 - 2) z_i \tag{34}$$

$$\hat{b} = \frac{1}{6} \sum_{i=1}^{9} d_i z_i \tag{35}$$

$$\hat{c} = \Delta^{-1} \sum_{i=1}^{9} (2\beta - 2d_i^2) z_i \tag{36}$$

The expression of the distance of the axis of the paraboloid (i.e. of the estimate of the edge) from the centre of the 3x3 neighbourhood is then:

$$\delta = -\frac{3\beta - 2}{2} \frac{\sum_{i=1}^{9} d_i z_i}{\sum_{i=1}^{9} (3d_i^2 - 2) z_i} \tag{37}$$

The expression for the normalized error of fit is:

$$\epsilon = \frac{||\mathbf{z} - \hat{\mathbf{z}}||}{||\mathbf{z}||} = 1 - \frac{||\hat{\mathbf{z}}||}{||\mathbf{z}||} = 1 - \frac{||A\hat{\mathbf{a}}||}{||\mathbf{z}||} \tag{38}$$

4 Experiments

We have tested our algorithm on a number of images and experimented varying systematically the parameters of the edge detector.

The filters in the implementation were constructed as described in [14, 11]. We give here a brief description:

- The even symmetric and odd-symmetric kernels were obtained by taking derivatives and Hilbert transforms of elongated Gaussian functions.
- The base Gaussian $G(x, y, \sigma_x, \sigma_y) = \frac{1}{2\pi\sqrt{\sigma_x \sigma_y}} \exp - \left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2} \right)$ was elongated along the horizontal (X) axis, therefore $\sigma_x \geq \sigma_y$.
- In this paper the ratio $\sigma_x : \sigma_y$ is called 'elongation', while the 'width' σ_y is called 'sdev' or 'standard deviation'.

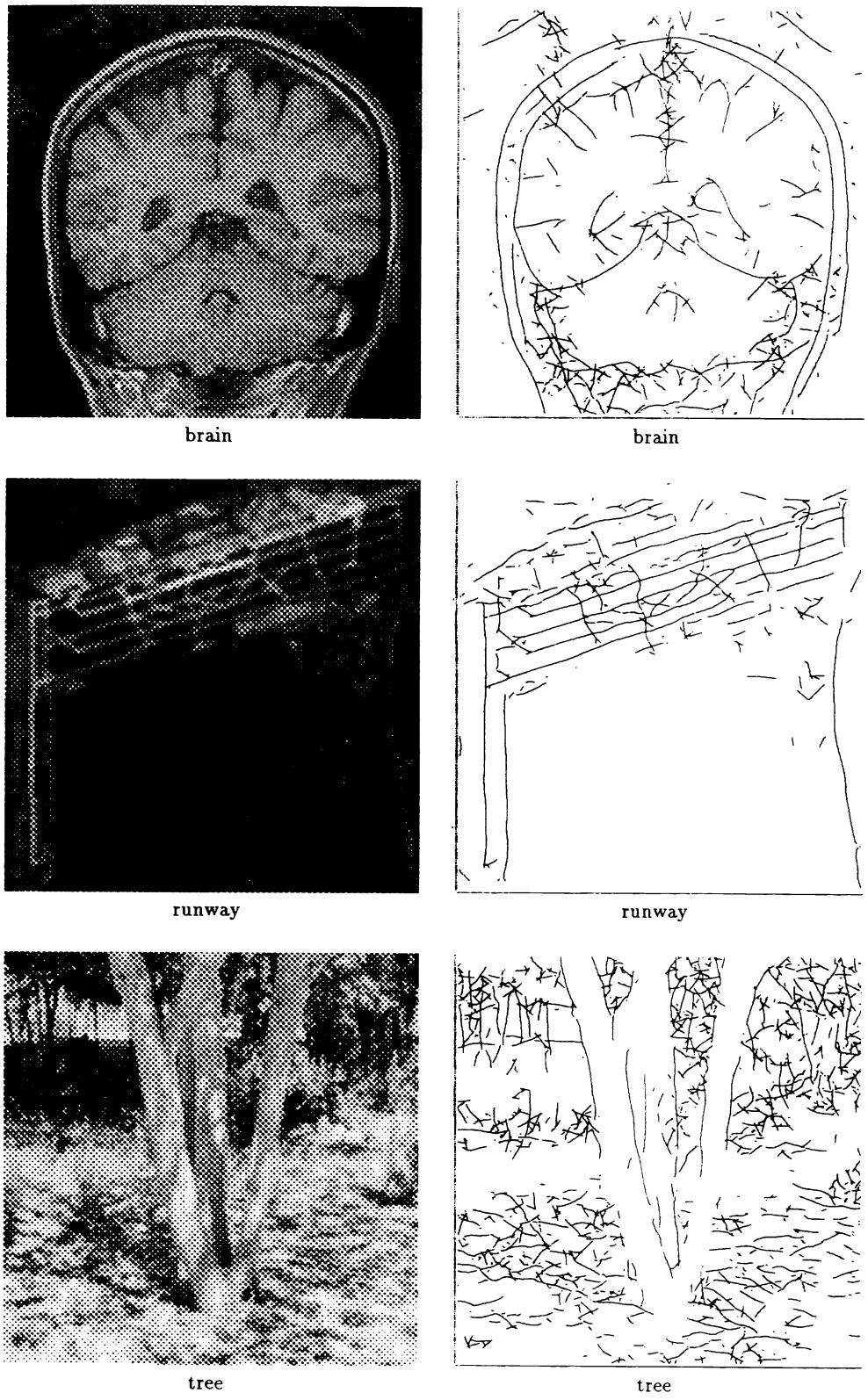
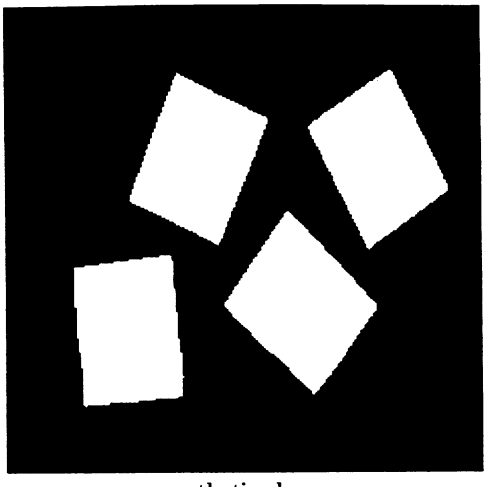
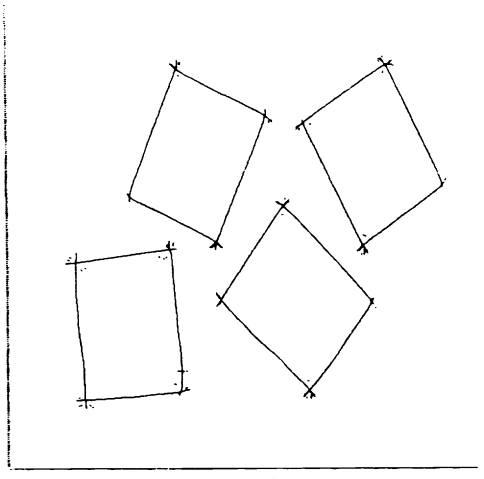


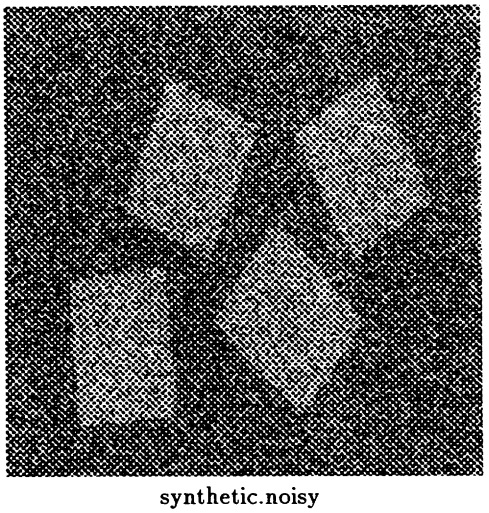
Fig. 3. The results of the experiment on 3 images. Kernel type: G2, sdev=1.2, elongation=3.0, energy thresh=0.01, 0.05, 0.05 (top to bottom).



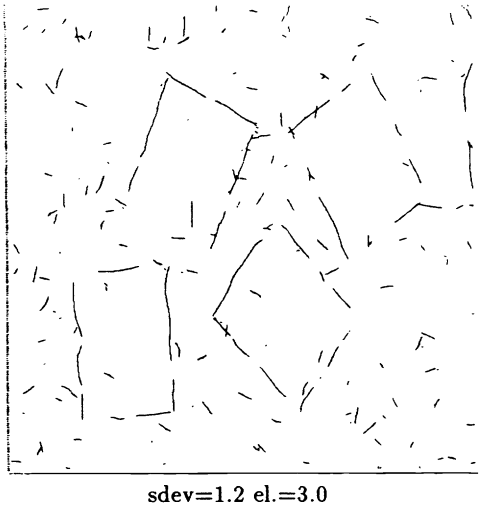
synthetic.clean



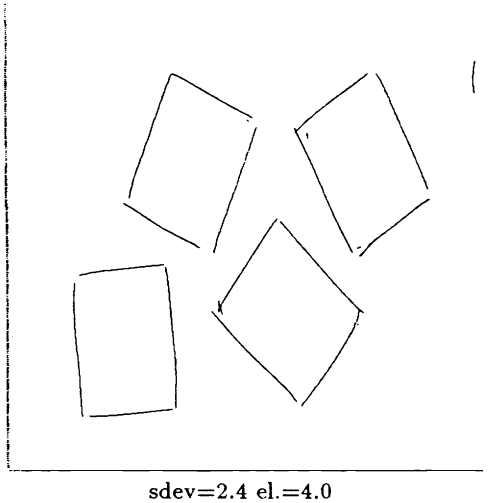
synthetic.clean



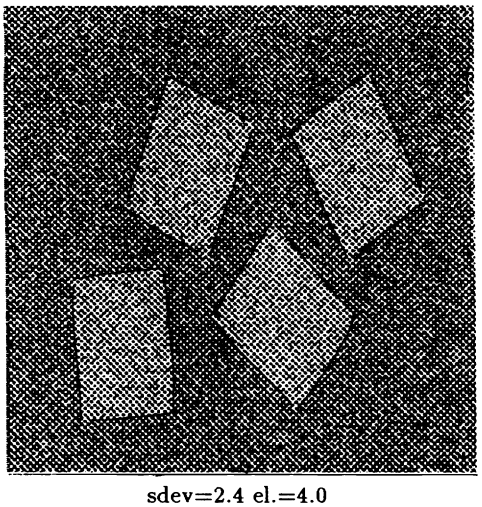
synthetic.noisy



sdev=1.2 el.=3.0



sdev=2.4 el.=4.0



sdev=2.4 el.=4.0

Fig. 4. Top and middle: The results of the experiment on 2 images. Kernel type: G2, sdev=1.2, elongation=3.0, energy thresh=0.01, 0.25. Bottom: using a more elongated and wider kernel improves dramatically the performance in presence of noise. Kernel type: G1, sdev=2.4, elongation=4.0, energy thresh=0.25. At left the edges. Edges superimposed to image at right.

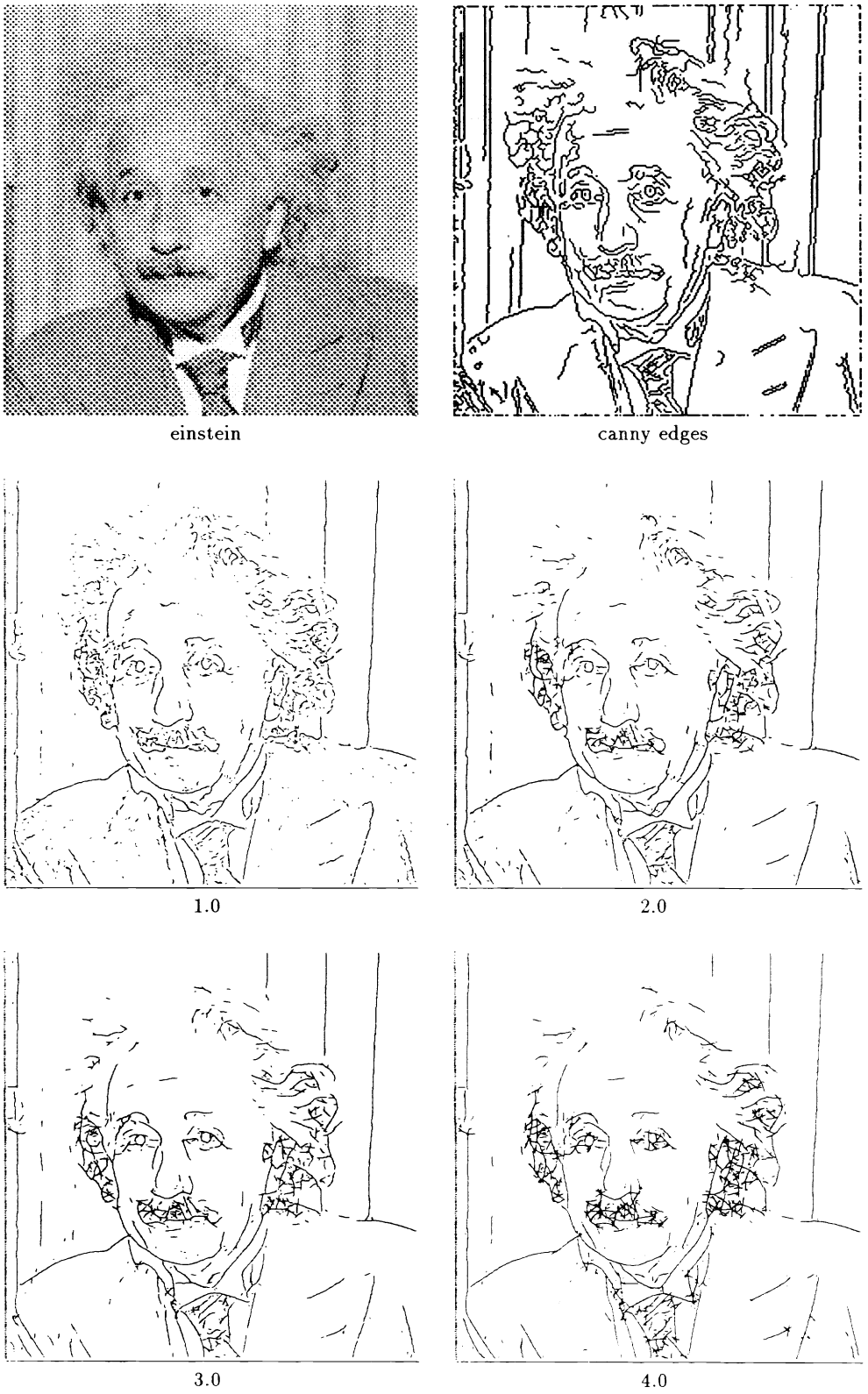


Fig. 5. Experiment with variable elongation. Top left, original figure:einstein, top right Canny edges, sdev = 0.7. Middle and Bottom. Filter type = G1, sdev = 0.7, elongation = 1.0, 2.0, 3.0, 4.0, Energy thresh.= 0.005.

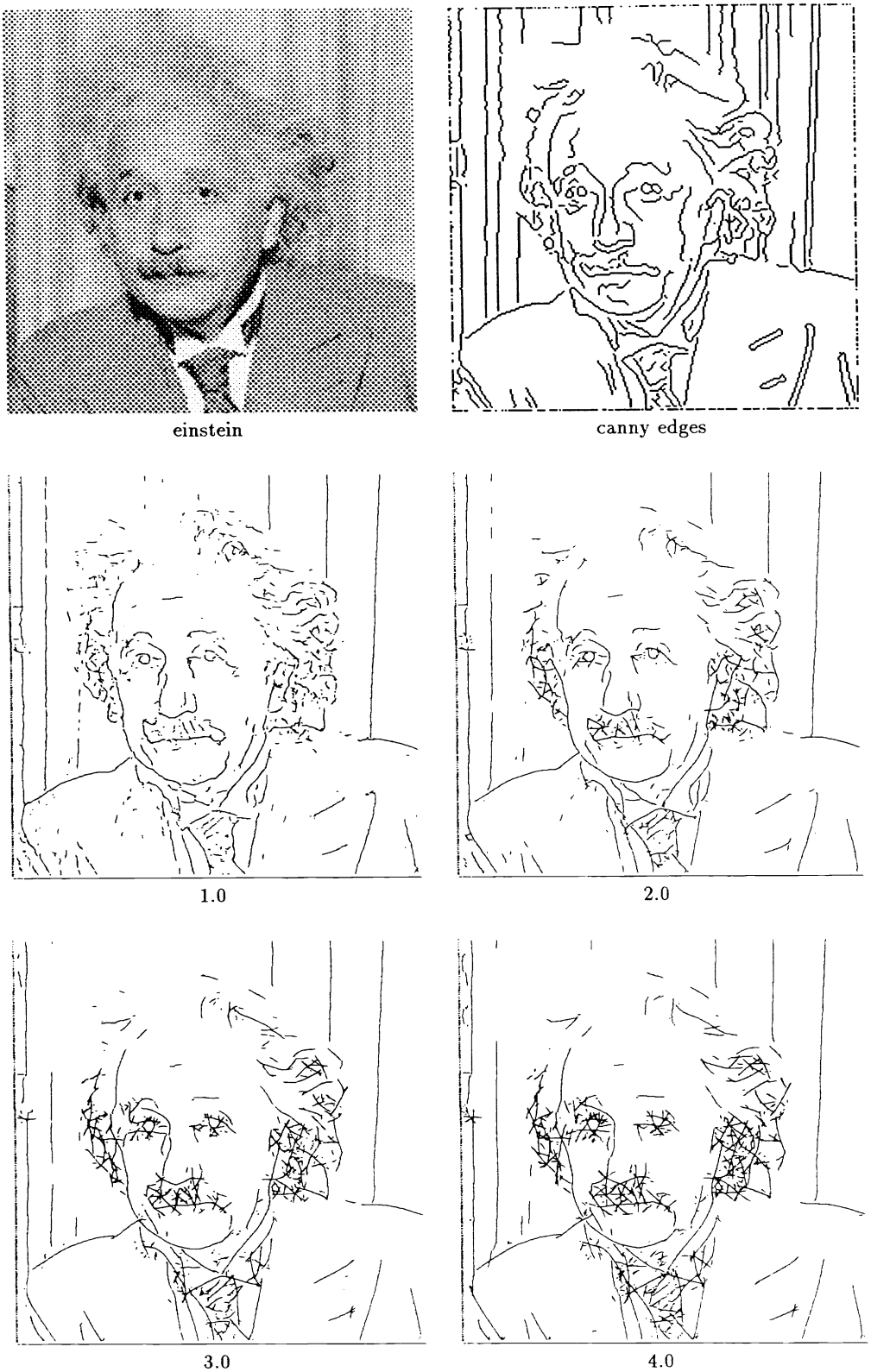


Fig. 6. Experiment with variable elongation. Top left, original figure:einstein, top right Canny edges, sdev = 1.2. Middle and Bottom. Filter type = G2, sdev = 1.2, elongation = 1.0, 2.0, 3.0, 4.0, Energy thresh.= 0.005.

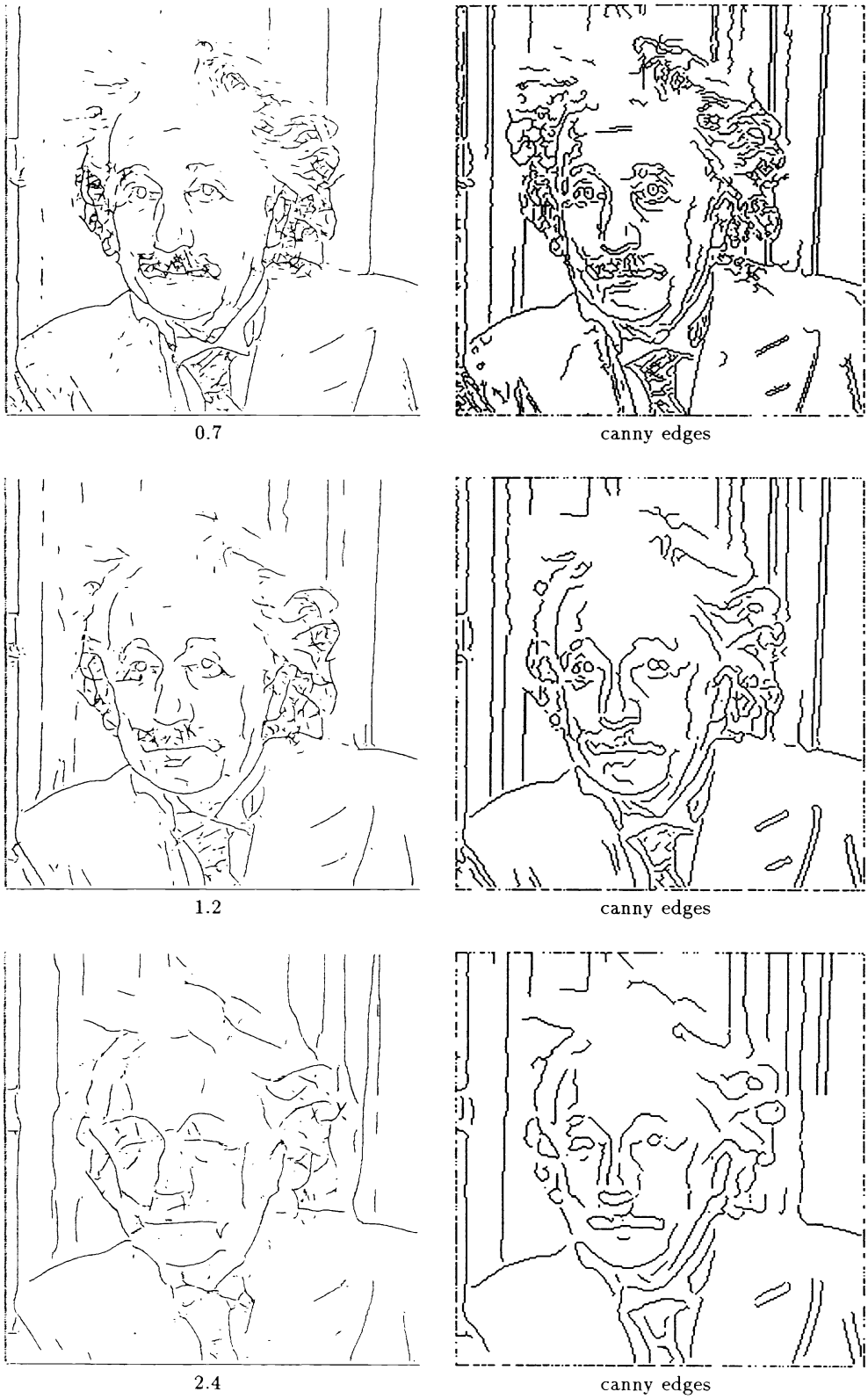


Fig. 7. Experiment with variable scale, original figure:einstein. Left Perona-Malik Right Canny edges. Sdev = 0.7, 1.2, 2.4. Filter type = G1, elongation = 2.0. Energy thresh.= 0.005.

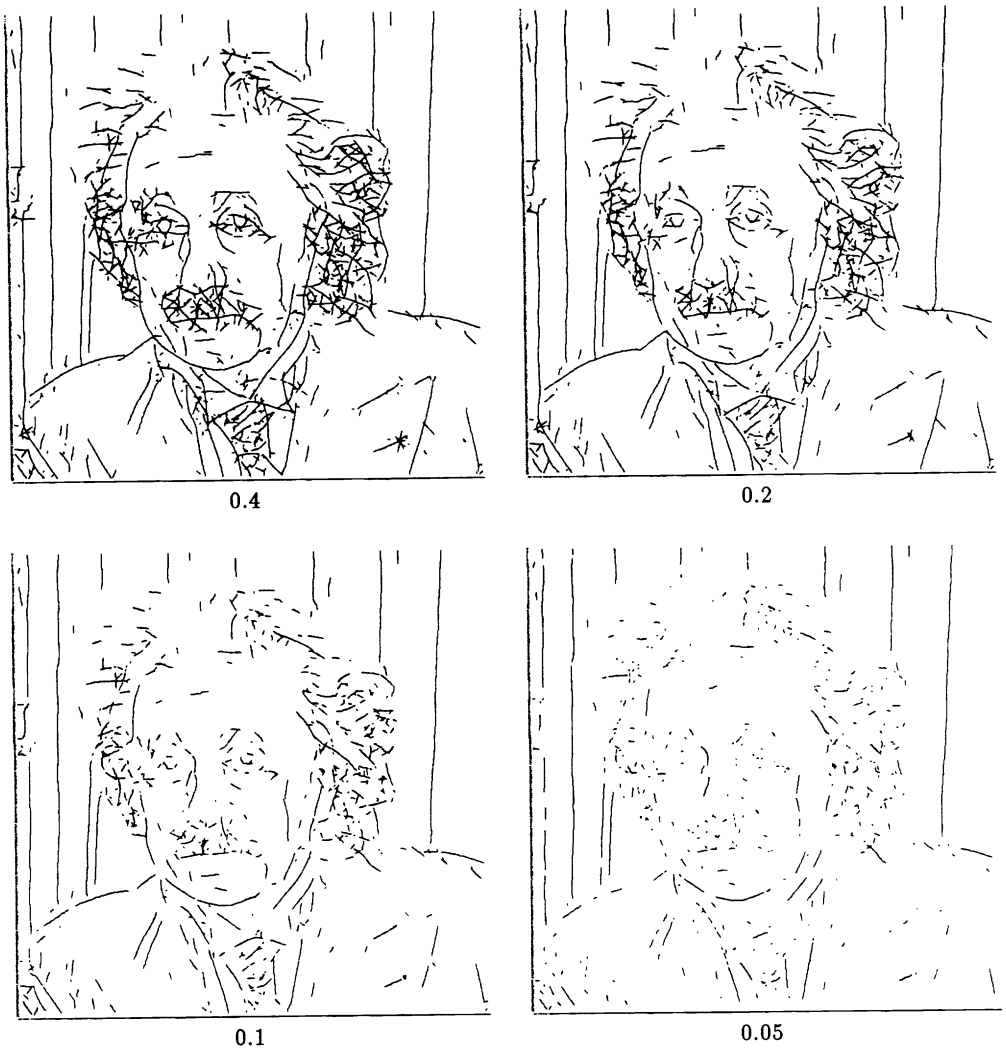


Fig. 8. Experiment with variable threshold on parabolic fit. Filter type = G2, sdev = 1.2, elongation = 3.0, parbola fit = 0.4, 0.2, 0.1, 0.05. Energy thresh.= 0.0025.

- The kernels called 'G1' were obtained by taking the 1st derivative along Y of the base Gaussian as the odd-symmetric kernel, and the Hilbert transform of it as the even-symmetric kernel.
- The kernels called 'G2' were obtained by taking the 2nd derivative along Y of the base Gaussian as the even-symmetric kernel, and its Hilbert transform as an odd kernel. In both cases the hilbert transforms were windowed by multiplication with a Gaussian function of width = 3 * sdev.

There were two thresholds used: (1) a contrast threshold ('energy thresh' in the captions) – no histheresis scheme was used, (2) a threshold on the fitting error in the subpixel localization stage (parabolic fit error). This was set to 60% for all experiments (i.e. edges were not marked at maxima where the fit had an error $\epsilon \geq 60\%$) besides the last experiment where this was varied systematically.

We performed four sets of experiments:

4.1 Standard images

Figures 3 and 4 show the output of the edge detector for five standard images. The edge detector had the same setting for all images, except for the contrast threshold which was chosen so as to obtain a reasonable amount of detail each time. The values of the parameters, including the contrast threshold, are in the fiture captions.

The experiment on the noisy image (Fig. 4) was repeated with a different set of kernels: larger and more elongated to achieve better SNR. Notice the fact that despite the large (sdev = 2.4 pixels) kernels there is no rounding of the corners thanks to the high orientation selectivity of the elongated kernels.

4.2 Variable elongation and kernel family

Figures 5 and 6 show experiments run on a sixth image using kernels the elongation of which was varied systematically. The kernels used were the G1 in Fig. 5, and the G2 in Fig. 6. The parameters of the detector are indicated in the figure captions. In each case the output of the Canny edge detector are shown for comparison. Notice that the higher the elongation of the kernels in the detector the better straight lines are recovered with respect to curved lines, and the better corners and junctions are recovered.

4.3 Variable scale

Figure 7 shows the results of an experiment in which the size of the kernels was veried systematically. The results are compared to the output of the Canny detector. Notice that the blurry vertical lines of the background are more easily detected at coarse scales. Notice also that our detector does not deform the shape of the boundaries at coarse scales as much as the Canny does.

4.4 Parabolic fit

In the last experiment we varied the threshold of parabolic fit, ϵ, see Fig. 8. The setting of the parameters was as in Fig. 6, with a lower (i.e. less demanding) contrast threshold (notice that now the vertical lines in the background are detected), and lower (i.e. more demanding) parabolic fit threshold. Notice that decreasing the parabolic fit threshold selects the straighter edges independently from the contrast.

References

1. T.O. Binford. Inferring surfaces from images. *Artificial Intelligence*, 17:205–244, 1981.
2. A. Blake and A. Zisserman. *Visual reconstruction.* MIT press, 1987.
3. J. Canny. A computational approach to edge detection. *IEEE trans. Pattern Anal. Mach. Intell.*, 8:679–698, 1986.
4. D. Forsyth and A. Zisserman. Mutual illumination. In *Proceedings of the IEEE CVPR*, pages 466–473, 1989.
5. W. Freeman and E. Adelson. Steerable filters for image analysis. Technical Report 126, MIT, Media Laboratory, 1990.
6. W. Freeman and E Adelson. The design and use of steerable filters for image analysis, enhancement and multi-scale representation. *IEEE trans. Pattern Anal. Mach. Intell.*, 1991.
7. B.K.P. Horn. Image intensity understanding. *Artificial intelligence*, 8(2):201–231, 1977.
8. M.C. Morrone and D.C. Burr. Feature detection in human vision: a phase dependent energy model. *Proc. R. Soc. Lond. B*, 235:221–245, 1988.
9. M.C. Morrone and R.A. Owens. Feature detection from local energy. *Pattern Recognition Letters*, 6:303–313, 1987.
10. P. Perona. Finite representation of deformable functions. Technical Report 90-034, International Computer Science Institute, 1947 Center st., Berkeley CA 94704, 1990.
11. P. Perona. Deformable kernels for early vision. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 222–227, June 1991.
12. P. Perona. Deformable kernels for early vision. Technical Report 2039, LIDS-MIT, October 1991. Submitted to IEEE PAMI.
13. P. Perona and J. Malik. Detecting and localizing edges composed of steps, peaks and roofs. Technical Report UCB/CSD 90/590, Computer Science Division (EECS), U.C.Berkeley, 1990.
14. P. Perona and J. Malik. Detecting and localizing edges composed of steps, peaks and roofs. In *Proceedings of the Third International Conference of Computer Vision*, pages 52–57. IEEE Computer Society, Osaka, 1990.
15. J. Ponce and M. Brady. Towards a surface primal sketch. Technical Report 824, MIT Artificial Intelligence Laboratory, 1985.
16. W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes in C.* Cambridge University Press, 1988.
17. S.O. Rice. Mathematical analysis of random noise. *Bell System Technical Journal*, 24:46–156, 1945.